INTERNATIONAL STANDARD

ISO/IEC 24824-3

First edition
2008-05-01

Information technology — Generic applications of ASN.1: Fast infoset security

Technologies de l'information — Applications génériques de l'ASN.1: Sécurité d'Infoset rapide

Reference number
ISO/IEC 24824-3:2008(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

CONTENTS

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 24824-3 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*, in collaboration with ITU-T. The identical text is published as ITU-T Rec. X.893 (05/07).

Introduction

This Recommendation | International Standard specifies:

a) the application of integrity to one or more parts of an XML infoset using Fast Infoset serialization and W3C XML Signature;

b) the application of encryption to one or more parts of an XML infoset using Fast Infoset serialization and W3C XML Encryption.

W3C XML Signature specifies a means of generating W3C XML Signature information items that contain (*inter alia*):

a) explicit (using URIs) or implicit (dependent on the use of the XML infoset signature information item) identification of one or more data objects (a data object is anything that either already is, or can be transformed into, a string of octets);

b) a (possibly empty) list of sequential transforms (specified by URIs for the algorithm to be used in performing the transform) from those data objects to a sequence of octets; these transforms can select all or part of the identified data objects, but are required to result in a sequence of octets;

c) digest and encryption information for the production of a signature of the resulting sequence of octets; and

d) the resulting signature.

This Recommendation | International Standard specifies four (canonical Fast Infoset) algorithms that can be referenced in a W3C XML Signature transform (and provides URIs for them) and can also be (independently) used as the algorithm for the W3C XML Signature canonicalization method.

NOTE 1 – The same Fast Infoset algorithm could be used for both the transform and the canonicalization method, but use of two different Fast Infoset algorithms (or a Fast Infoset algorithm and some other algorithm) is not excluded.

In all four cases, the input to the canonical Fast Infoset algorithm is either an XML infoset, or an XPath node set (restricted, in accordance with 6.1.4 b, to those node sets that produce a well-formed XML document when serialized).

The output of all four canonical Fast Infoset algorithms is a sequence of octets (the octets of a fast infoset document, see ITU-T Rec. X.891 | ISO/IEC 24824-1) that are suitable for digest and hashing in order to provide a signature in accordance with W3C XML Signature.

NOTE 2 – This will usually be the last transform in the sequential list of W3C XML Signature transforms, but need not be.

A typical use will be to sign one or more parts of a single XML infoset.

NOTE 3 – Use to sign parts of multiple XML infosets is not excluded.

It is expected, but not required, that the resulting W3C XML Signature information items will be used either as a detached signature, or as an enveloping or enveloped signature (see W3C XML Signature) for the XML infoset that is signed, and that the resulting XML infoset will be serialized using ITU-T Rec. X.891 | ISO/IEC 24824-1.

This Recommendation | International Standard specifies application-level extensions (see 3.2.1) to W3C XML Encryption. These application-level extensions enable encryption to be applied to part of an XML infoset using octets provided by a Fast Infoset serialization, rather than to the octets provided by an XML serialization of those parts.

NOTE 4 – W3C XML Encryption can be applied to a complete fast infoset document as specified in W3C XML Encryption, 3.1, without the use of this Recommendation | International Standard. The **MimeType** attribute will have the value **"application/fastinfoset"**.

The means of identifying the parts of the XML infoset that are encrypted is specified by W3C XML Encryption and allows the encryption of:

a) an element information item and its properties, including any direct or indirect child information items (and their properties); and

b) the child information items of the child property of an element information item and their properties, including any direct or indirect child information items (and their properties).

Encryption requires that those parts of an XML infoset that are to be encrypted have to be first serialized into a string of octets for input to an encryption algorithm.

The ability to produce a serialization of a and b above is not supported by ITU-T Rec. X.891 | ISO/IEC 24824-1, but is specified in clause 8 of ITU-T Rec. X.893 | ISO/IEC 24824-3 (using ITU-T Rec. X.891 | ISO/IEC 24824-1). This is done by converting such fragments (in a defined way) to a complete XML infoset and then applying ITU-T Rec. X.891 | ISO/IEC 24824-1 to the complete XML infoset.

This Recommendation | International Standard also specifies two URIs, one for a above and one for b above, that are used in XML Encryption to identify the application-level extensions which determine the use of Fast Infoset serialization rather than XML serialization for the production of the octets to be input to an encryption algorithm.

Use of Fast Infoset serialization to determine the octets for input to an encryption algorithm in general reduces the number of octets that have to be encrypted and decrypted, and would be normal (but not necessary) if the XML infoset is transferred using a Fast Infoset serialization.

NOTE 5 – It is also possible (but would be unusual) to use Fast Infoset serialization to determine the octets for input to an encryption algorithm when the XML infoset is to be transferred using an XML serialization.

The serialization of an XML infoset containing W3C XML Signature information items and/or W3C XML Encryption information items to a fast infoset document has the following advantages over serialization to an XML document:

a) repeating information such as multiple signed references or multiple encrypted parts with the same XML tags or content will be encoded more efficiently; and

b) the (binary) octets associated with signature values, digest values, cipher values or keys may be encoded directly (see ITU-T Rec. X.891 | ISO/IEC 24824-1, 10.3) if a (binary) fast infoset document is used to serialize the XML infoset; when serializing an XML infoset to an XML document (which is a string of characters), such octets are required to be base64 encoded, increasing processing speed and size.

Clause 6 specifies four canonical Fast Infoset algorithms that can be referenced in a W3C XML Signature transform.

Clause 7 specifies the use of W3C XML Signature with canonical Fast Infoset algorithms.

Clause 8 specifies the use of W3C XML Encryption for the encryption of parts of an XML infoset that are serialized to fast infoset documents.

Annex A does not form an integral part of this Recommendation | International Standard and provides examples of signing and validating a SOAP XML infoset (that makes use of canonical Fast Infoset algorithms), and encrypting and decrypting a SOAP message infoset (that makes use of the encryption of part of the SOAP message infoset that is serialized to a fast infoset document).

Annexes B and C do not form an integral part this Recommendation | International Standard, and provide examples of a signed SOAP message infoset and a signed and encrypted SOAP message infoset, respectively.

INTERNATIONAL STANDARD
ITU-T RECOMMENDATION

Information technology – Generic applications of ASN.1: Fast infoset security

1 Scope

This Recommendation | International Standard specifies four (canonical Fast Infoset) algorithms that can be used in the application of W3C XML Signature (and provides URIs for them).

It also specifies application-level extensions to the W3C XML Encryption processing rules for the encryption of part of an XML infoset (see 8.1) serialized as a fast infoset document and for the decryption of an encrypted part (see 8.3) that was serialized as a fast infoset document.

The use of any resulting W3C XML Signature information items or W3C XML Encryption information items is not within the scope of this Recommendation | International Standard.

2 Normative references

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations. The IETF maintains a list of RFCs, together with those that have been obsoleted by later RFCs. The reference to a document within this Recommendation | International Standard does not give it, as a stand-alone document, the status of a Recommendation or International Standard.

2.1 Identical Recommendations | International Standards

– ITU-T Recommendation X.891 (2005) | ISO/IEC 24824-1:2007, *Information technology – Generic applications of ASN.1: Fast infoset*.

2.2 Additional references

– ISO/IEC 10646:2003, *Information technology – Universal Multiple-Octet Coded Character Set (UCS)*.

– W3C Canonical XML:2001, *W3C Canonical XML Version 1.0, W3C Recommendation, Copyright © [15 March 2001] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2001/REC-xml-c14n-20010315*.

– W3C XML Encryption:2002, *XML Encryption Syntax and Processing, W3C Recommendation, Copyright © [10 December 2002] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2002/REC-xmlenc-core-20021210*.

– W3C Exclusive Canonical XML:2002, *W3C Exclusive XML Canonicalization Version 1.0, W3C Recommendation, Copyright © [18 July 2002] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2002/REC-xml-exc-c14n-20020718.*

– W3C XML Information Set:2004, *XML Information Set (Second Edition), W3C Recommendation, Copyright © [04 February 2004] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2004/REC-xml-infoset-20040204.*

– W3C XML Signature:2002, *XML-Signature Syntax and Processing, W3C Recommendation, Copyright © [12 February 2002] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2002/REC-xmldsig-core-20020212*.

– W3C XPath:1999, *XML Path Language (XPath) Version 1.0, W3C Recommendation, Copyright © [16 November 1999] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/1999/REC-xpath-19991116.*

3 Definitions

For the purposes of this Recommendation | International Standard, the following definitions apply.

3.1 Imported definitions

This Recommendation | International Standard uses the following terms defined in ITU-T Rec. X.891 | ISO/IEC 24824-1:

a) fast infoset document;

b) information item;

c) initial vocabulary;

d) XML infoset.

3.2 Additional definitions

3.2.1 application-level extensions (for W3C Encryption): A term applied to requirements of this Recommendation | International Standard that specify the actions to be taken by an application when the W3C Encryption processing rules do not fully specify the actions to be taken.

3.2.2 canonical Fast Infoset algorithm: An algorithm that takes as input an XML infoset (see W3C XML Information Set) or an XPath node set (see W3C XPath) and generates, as output, a canonical fast infoset document.

3.2.3 canonical fast infoset document: A fast infoset document generated by a canonical Fast Infoset algorithm.

3.2.4 canonical XML algorithm: An algorithm that takes as input an XML infoset, a well-formed XML document or an XPath node set, and generates, as output, a well-formed XML document in canonical form.

NOTE – Canonical XML algorithms are currently specified by W3C Canonical XML and W3C Exclusive Canonical XML.

3.2.5 canonical XML document: A well-formed XML document generated by a canonical XML algorithm.

3.2.6 element part (of an XML infoset): An **element** information item (and all information items that are descendents of the **element** information item).

3.2.7 element content part (of an XML infoset): All the information items in the **[children]** property of an **element** information item (and all information items that are descendents of those information items).

4 Abbreviations

For the purposes of this Recommendation | International Standard, the following abbreviations apply:

URI	Uniform Resource Identifier
UTF-8	Universal Transformation Function 8-bit (see ISO/IEC 10646, Annex D)
W3C	World Wide Web Consortium

5 Notation

5.1 In this Recommendation | International Standard, **`bold Courier`** is used for ASN.1 notation and **bold Arial** is used for W3C XML syntax and for the names of information items of the XML Information Set.

5.2 The names of information items' properties are in **bold Arial** and enclosed between square brackets (for example, **[children]** property).

5.3 URIs are in **bold Arial** and enclosed between normal quotes.

EXAMPLE: "**http://www.w3.org/2003/05/soap-envelope**".

6 Canonical Fast Infoset algorithms

6.1 Requirements on canonical Fast Infoset algorithms

6.1.1 The following subclauses specify the input to and the general requirements on canonical Fast Infoset algorithms. Specific Fast Infoset algorithms (that make reference to this clause) are specified in 6.4.

6.1.2 A canonical Fast Infoset algorithm shall specify a canonical XML algorithm that is used in the conceptual transformation process (see 6.1.5).

NOTE – It is not in the scope of this Recommendation | International Standard to specify canonical XML algorithms. Algorithms used in 6.4 are currently (only) specified in W3C Canonical XML and W3C Exclusive Canonical XML.

6.1.3 A canonical Fast Infoset algorithm shall specify a URI that is used for declaration of the algorithm in W3C XML Signature information items (see 7.2).

6.1.4 A canonical Fast Infoset algorithm shall produce a canonical fast infoset document by the transformation of either or both of the following inputs:

a) an XML infoset; or

b) an XPath node set that produces a well-formed XML document when transformed as specified in 6.1.5 a.

NOTE 1 – Support of XPath node sets is required to ensure that this Recommendation | International Standard is compatible with the XML security-related standards. W3C Canonical XML and W3C Exclusive Canonical XML use the XPath data model (see W3C XPath, clause 5). W3C XML Signature specifies transformations using XPath for canonicalization and filtering (see W3C XML Signature, 6.5 and 6.6.3 respectively).

NOTE 2 – Input of an XPath node set that produces an XML document that is not well-formed is not supported for Fast Infoset canonicalization (and hence for W3C XML Signatures produced using a canonical Fast Infoset algorithm).

6.1.5 The conceptual transformation steps performed by a canonical Fast Infoset algorithm to produce a canonical fast infoset document shall be as follows:

a) the input XML infoset or input XPath node set is transformed (by a canonical XML algorithm) to produce a canonical XML document, as specified in 6.2;

b) the canonical XML document is parsed to produce an XML infoset; this will be a canonical XML infoset; and

c) the canonical XML infoset is serialized as a canonical fast infoset document, with restrictions specified in 6.3.

NOTE – Implementations may choose to optimize the steps so that an XML infoset or an XPath node set is transformed directly to a canonical fast infoset document without producing the intermediate canonical XML document as long as the result is the same as if all steps were performed.

6.1.6 When serializing into a canonical fast infoset document, the order of attributes shall be the order of the corresponding Canonical XML document.

NOTE 1 – **Attribute** information items among the **[namespace attributes]** and **[attributes]** properties of **element** information items are unordered (see W3C XML Information Set, 2.2). Subclause 6.1.6 preserves the document order of attributes information items produced from parsing the canonical XML document.

NOTE 2 – W3C Canonical XML (an XML canonical algorithm) extends the document order of XPath node sets (see W3C XPath, 5) such that an element's namespace and attribute nodes are canonically ordered (see W3C Canonical XML, 2.2).

6.2 Requirements on canonical XML algorithms for use by a canonical Fast Infoset algorithm

6.2.1 The following subclause specifies the requirements that a canonical XML algorithm has to satisfy in order for it to be used when defining a canonical Fast Infoset algorithm.

NOTE – The algorithms specified in W3C Canonical XML and W3C Exclusive Canonical XML satisfy these requirements.

6.2.2 A canonical XML algorithm used in defining a canonical Fast Infoset algorithm shall be capable of transforming (to a well-formed canonical XML document) all those inputs that the canonical Fast Infoset algorithm supports (see 6.1.4).

NOTE – Such canonical XML algorithms are specified by W3C Canonical XML and W3C Exclusive Canonical XML.

6.3 Restrictions when serializing an XML infoset to a canonical fast infoset document

NOTE – This serialization is step c of 6.1.5 which is used when producing octets for signing.

6.3.1 Values of the **`NonIdentifyingStringOrIndex`** type (see ITU-T Rec. X.891 | ISO/IEC 24824-1, 7.14) shall consist of the **`literal-character-string`** alternative with the **`add-to-table`** component set to **`FALSE`**.

6.3.2 The UTF-8 encoding (see ISO/IEC 10646) shall be used for all character strings represented as values of the **`EncodedCharacterString`** type (see ITU-T Rec. X.891 | ISO/IEC 24824-1, 7.17).

NOTE – Such character strings will be associated with sequences of adjacent character information items and the **[normalized value]** properties of **attribute** information items.

6.3.3 A sequence of adjacent character information items, starting from the first character information item that has no previous character item directly next to it in the **[children]** property to the last character information that has no further character information item directly next to it in the **[children]** property, shall be represented by a single value of the **`CharacterChunk`** type (see ITU-T Rec. X.891 | ISO/IEC 24824-1, 7.7).

6.3.4 If the sequence of adjacent character information items exceeds the maximum allowed for a value of the **`CharacterChunk`** type (2^{32}), then there shall be consecutive values of a **`CharacterChunk`** type for each consecutive maximum sequence of adjacent character information items.

6.3.5 A canonical fast infoset document shall not have an initial vocabulary. The **`initial-vocabulary`** component of a value of the **`Document`** type shall be absent (see ITU-T Rec. X.891 | ISO/IEC 24824-1, 7.2.1).

6.3.6 A vocabulary table (see ITU-T Rec. X.891 | ISO/IEC 24824-1, clause 6) shall not contain duplicate table entries. ITU-T Rec. X.891 | ISO/IEC 24824-1, 7.13.7, is applied with the restriction that action 7.13.7 b shall not be performed if an identical character string exists in the current content of the applicable string table.

NOTE – The CONTENT CHARACTER CHUNK table and the ATTRIBUTE VALUE table (see ITU-T Rec. X.891 | ISO/IEC 24824-1, 8.4) will contain no table entries due to the restriction specified in 6.3.1.

6.4 Canonical Fast Infoset algorithms

6.4.1 The following subclauses specify four canonical Fast Infoset algorithms. In each case, the canonical XML algorithm to be used is specified (see 6.1.2), together with the URI for the Fast Infoset algorithm (see 6.1.3).

6.4.2 The "inclusive canonical Fast Infoset algorithm without comments" shall be identified by the URI "**urn:fastinfoset:c14n:inclusive**" using the canonical XML algorithm specified in W3C Canonical XML, with the second input parameter (see W3C Canonical XML, 2.1) set to false.

NOTE – The second input parameter is a boolean that indicates whether or not comments should be included in the canonical form produced by the Canonical XML algorithm.

6.4.3 The "inclusive canonical Fast Infoset algorithm with comments" shall be identified by the URI "**urn:fastinfoset:c14n:inclusive:withcomments**" using the canonical XML algorithm specified in W3C Canonical XML, with the second input parameter (see W3C Canonical XML, 2.1) set to true.

6.4.4 The "exclusive canonical Fast Infoset algorithm without comments" shall be identified by the URI "**urn:fastinfoset:c14n:exclusive**" using the canonical XML algorithm specified in W3C Exclusive Canonical XML, with the second input parameter (see W3C Exclusive Canonical XML, clause 3) set to false. This Fast Infoset algorithm has a parameter that is an "InclusiveNamespace PrefixList" parameter (see W3C Exclusive Canonical XML, 1.1), which can be null, and which is passed unmodified to the canonical XML algorithm.

6.4.5 The "exclusive canonical Fast Infoset algorithm with comments" shall be identified by the URI "**urn:fastinfoset:c14n:exclusive:withcomments**" using the canonical XML algorithm specified in W3C Exclusive Canonical XML, with the second input parameter (see W3C Exclusive Canonical XML, clause 3) set to true. This Fast Infoset algorithm has a parameter that is an "InclusiveNamespace PrefixList" parameter (see W3C Exclusive Canonical XML, 1.1), which can be null, and which is passed unmodified to the canonical XML algorithm.

7 W3C XML Signature and Fast Infoset

7.1 The use of a canonical Fast Infoset algorithm (see 6.4) is specified in the following subclauses.

7.2 The **Algorithm attribute** information item in the **[attributes]** property of a **CanonicalizationMethod element** information item (see W3C XML Signature, 4.3.1) or a **Transform element** information item (see W3C XML Signature, 4.3.3.4) shall have a **[normalized value]** property that is a URI identifying a canonical Fast Infoset algorithm (see 6.1.3).

7.3 If the canonical Fast Infoset algorithm specifies the canonical XML algorithm (see 6.1.2) by reference to W3C Exclusive Canonical XML and an "InclusiveNamespace PrefixList" parameter (see W3C Exclusive Canonical XML, 1.1) is given as input (see 6.4.4 and 6.4.5), then the parameter shall be represented as specified in W3C Exclusive Canonical XML, clause 4.

8 W3C XML Encryption and Fast Infoset

W3C XML Encryption permits (and this Recommendation | International Standard supports) the encryption of element parts and element contents parts of an XML infoset.

8.1 Application-level extensions for encryption

8.1.1 Each data item (see W3C XML Encryption, 4.1) to be encrypted shall be an element part or an element content part of an XML infoset, selected by the encrypting application.

8.1.2 The encryption processing operations (specified by W3C XML Encryption, 4.1) to a part of an XML infoset shall be extended for the operations 3.2, 4 and 5.2 of W3C XML Encryption, 4.1, as specified in the three following subclauses.

8.1.3 Operation 3.2 of W3C XML Encryption, 4.1, shall be extended to obtain the octets to be encrypted as follows:

a) the selected part of the original XML infoset (A) shall be converted to a complete XML infoset (B) as specified in 8.2;

b) that XML infoset (B) shall be serialized using ITU-T Rec. X.891 | ISO/IEC 24824-1, with the restriction that no external vocabulary shall be used; and

c) the resulting octets shall be the octets to be encrypted in operation 3.3 of W3C XML Encryption, 4.1.

8.1.4 Operation 4 of W3C XML Encryption, 4.1, shall be extended to include a **Type attribute** information item in the **[attributes]** property of the **EncryptedData element** information item (see W3C XML Encryption, 3.1) whose **[normalized value]** property shall be one of the following:

a) if the part of the XML infoset is an element part, then the **[normalized value]** of the **Type attribute** information item shall be the URI "**urn:fastinfoset:element**"; or

b) if the part of the XML infoset is an element content part, then the **[normalized value]** of the **Type attribute** information item shall be the URI "**urn:fastinfoset:element-content**".

8.1.5 Operation 5.2 of W3C XML Encryption, 4.1, shall be extended such that the **EncryptedData element** information item (produced by operation 4 of W3C XML Encryption, 4.1, extended as specified in 8.1.4) shall replace the part of the XML infoset that was processed in operation 3 (extended as specified in 8.1.3).

8.2 Generation of a complete XML infoset from part of an XML infoset

8.2.1 Generation from an element part of an XML infoset

8.2.1.1 A complete XML infoset shall be generated with a **document** information item that has the following properties:

a) a **[children]** property whose only member is a copy (E, say) of the **element** information item (and all its properties, including the **[children]** property) that is the element part of the original XML infoset that is to be encrypted; and

b) a **[document element]** property that is E.

8.2.1.2 The **[namespace attributes]** property of E (see 8.2.1.1 a) shall be modified so that it is consistent with the **[in-scope namespaces]** property of E.

NOTE 1 – An implementation may choose to remove any unused namespace information items in the **[in-scope namespaces]** property of E (and its descendents) before the **[namespace attributes]** property of E is modified.

NOTE 2 – For further detail on the recommended handling of information items corresponding to default namespace declarations and XML-specific **attribute** information items, see W3C XML Encryption, 4.3.3.

8.2.2 Generation from an element content part of an XML infoset

8.2.2.1 A complete XML infoset shall be generated with a **document** information item that contains the following properties:

a) a **[children]** property whose only member is an **element** information item (E, say) as specified in 8.2.2.2; and

b) a **[document element]** property that is E.

8.2.2.2 The **element** information item E (see 8.2.2.1 a) shall have no value for the **[namespace name]** and **[prefix]** properties and shall have values for the following properties:

a) a **[local name]** property of **"content"**;

b) a **[children]** property that is a copy of the **[children]** property of the element content part of the original XML infoset; and

c) a **[namespace attributes]** property of E that is consistent with all the **[in-scope namespaces]** properties of the **element** information items among the **[children]** property of the element content part of the original XML.

NOTE 1 – An implementation may choose to remove any unused namespace information items in the **[in-scope namespaces]** properties of the **element** information items (and descendents) in the **[children]** property of the XML infoset fragment before the **[namespace attributes]** property of E is modified.

NOTE 2 – For further detail on the recommended handling of information items corresponding to default namespace declarations and XML-specific **attribute** information items, see W3C XML Encryption, 4.3.3.

8.3 Application-level extensions for decryption

Operation 5.0 of W3C XML Encryption, 4.2, shall be extended to process an **EncryptedData element** information item that contains a **Type attribute** information item (see W3C XML Encryption, 3.1) among the **[attributes]** property whose **[normalized value]** property is one of the URIs specified in 8.1.4. The following steps shall be performed:

a) the octet sequence obtained in operation 3 (see W3C XML Encryption, 4.2) shall be interpreted as a fast infoset document;

b) an XML infoset shall be generated by parsing that fast infoset document;

c) parts of that XML infoset shall be used to replace the **EncryptedData element** information item as follows:

1) if the URI is "**urn:fastinfoset:element**", then the **element** information item that is the **[document element]** property of the **document** information item of that XML infoset shall replace the **EncryptedData element** information item;

2) if the URI is "**urn:fastinfoset:element-content**", then all the information items in the **[children]** property of the **element** information item that is the **[document element]** property of the **document** information item of that XML infoset shall replace the **EncryptedData element** information item.

Annex A

Examples of signing and encrypting an XML infoset

(This annex does not form an integral part of this Recommendation | International Standard)

A.1 Introduction of examples

A.1.1 All XML infosets presented in this annex will be represented as XML documents. For reasons of clarity and brevity, namespaces URIs and textual content (that is base64 [IETF RFC 2045] encoded octets) that is not instructive to explanation will be truncated or completely removed and represented by the characters "...".

A.1.2 The cryptographic algorithms presented in this annex are for explanation only. This Recommendation | International Standard does not guarantee the use of such algorithms.

A.1.3 This annex presents two examples: the signing of an XML infoset (see A.2), and the signing and encrypting of an XML infoset (see A.3).

A.1.4 The XML infoset chosen in each example (to be signed or signed and encrypted) is the following SOAP message infoset:

```
<soap:Envelope xmlns:soap="...">
  <soap:Body>
    <n:payment xmlns:n="...">1000</n:payment>
  </soap:Body>
</soap:Envelope>
```

A.1.5 The resulting signed or signed and encrypted SOAP message infoset conforms to that specified by OASIS Web Services Security [WSS] and the WS-I Basic Profile [WS-I], with the exceptions that a canonical Fast Infoset algorithm is used for signing (see 6.4) and encrypted content is identified as a fast infoset document (see 8.1.4).

A.1.6 The examples are designed to present the processes by which an XML infoset may be signed/validated or signed/validated and encrypted/decrypted and therefore does not present the exact information for keys, signature values, digest values and cipher data.

A.1.6.1 The SOAP message infoset represents a simple example of a payment request of 1000 units by a client to a service. To ensure that the 1000 units do not get modified, for example by a "man in the middle" when the SOAP message is sent over a public network, the **soap:Body element** information item and contents can be signed and the receiver can validate that the contents were not tampered with. To ensure that only trusted parties can see the message (and that a payment of 1000 units has been requested), the **soap:Body element** information item and contents can be signed, and then the contents can be encrypted.

A.2 Signing and verifying the SOAP message infoset

A.2.1 The signed SOAP message infoset

A.2.1.1 The signed SOAP message infoset of the SOAP message infoset in A.1.4 where the SOAP body has been signed is presented in Annex B.

A.2.1.2 The signed SOAP message infoset utilizes a detached signature where the signed entity (the **soap:Body element** information item) and the signature (the **ds:Signature element** information item) are detached from each other.

A.2.2 Generating the signed SOAP message infoset

NOTE – The cryptographic algorithms used in this subclause are only explanatory. The use of SHA-1 has been deprecated by some standards bodies.

A.2.2.1 The process of signature generation is specified in W3C XML Signature, 3.1. A description of this process as applied to the example SOAP message infoset follows.

A.2.2.2 The signing application generates references to data objects that are to be signed, collects those references, and then generates the signed information of the digital signature.

A.2.2.3 The signing application selects that the single data object to be signed is the **soap:Body element** information item (and content) of the SOAP message infoset in A.1.4.

A.2.2.4 The signing application identifies the data object so that it can be referenced. The data object is identified by the addition of the **wsu:Id attribute** information item, on the **soap:Body element** information item, whose **[normalized value]** property is the identifier "**TheBody**".

A.2.2.5 The signing application selects that the transformations to calculate the digest value over the resulting data object consist of a single transform that specifies the "exclusive canonical Fast Infoset algorithm without comments" (see 6.4.4 and clause 7). This results in the production of the **ds:Transforms element** information item, that contains a single child **ds:Transform element** information item with an **Algorithm attribute** information item whose **[normalized value]** property is "**urn:fastinfoset:c14n:exclusive**".

A.2.2.6 The signing application selects the SHA-1 digest algorithm (see [FIPS 180-2]) to generate the digest value. The digest value is generated by applying the single transformation from the data object to a sequence of octets (which are then digested using the SHA-1 algorithm) as follows (see 6.1.5):

a) the input to the "exclusive canonical Fast Infoset algorithm without comments" is the XPath node set of the **soap:Body element** information item and its child information items of the SOAP message infoset in A.1.4 (see 6.1.5 a);

b) a canonical XML document is produced from the XPath node set (see 6.1.5 a) using the canonical XML algorithm specified in W3C Exclusive Canonical XML (see 6.4.4);

c) the canonical XML document is parsed to produce an XML infoset (see 6.1.5 b);

d) the canonical XML infoset is serialized as a canonical fast infoset document (see 6.1.5 c), with the restrictions on serialization specified in 6.3; and

e) the sequence of octets that is input to the SHA-1 digest algorithm is the canonical fast infoset document.

A.2.2.7 The generation of the digest value results in the production of the **ds:DigestMethod element** information item and the **ds:DigestValue element** information item. The element content of the **ds:DigestValue element** information item will contain the characters that are the base64 (see [IETF RFC 2045]) encoding of the 160-bit digest value generated by the SHA-1 digest algorithm.

A.2.2.8 The signing application generates the reference to the data object that is signed. This results in the production of a **ds:Reference element** information item that has a **URI attribute** information whose **[normalized value]** property is the URI "**#TheBody**" that references the data object. The **ds:Reference element** information item contains, as child element information items, the previously produced **ds:Transforms**, **ds:DigestMethod** and the **ds:DigestValue element** information items.

A.2.2.9 Next, the signing application collects the reference and generates the signed information.

A.2.2.10 The signing application selects the "exclusive canonical Fast Infoset algorithm without comments" (see 6.4.4 and clause 7) as the algorithm of canonicalization method. This results in the production of the **ds:CanonicalizationMethod element** information item with an **Algorithm attribute** information item whose **[normalized value]** property is "**urn:fastinfoset:c14n:exclusive**".

A.2.2.11 The signing application selects the RSA-SHA-1 signature algorithm (see [IETF RFC 3447]) as the signature method to calculate the signature value. This results in the production of the **ds:SignatureMethod element** information item.

A.2.2.12 The signing application generates the signed information to be signed using the signature method. This results in the production of the **ds:SignedInfo element** information item that contains, as child information items, the previously produced **ds:CanonicalizationMethod**, **ds:SignatureMethod** and the **ds:Reference element** information items.

A.2.2.13 The signing application selects the key to be used for signing (in this example an X.509 security token, see [ITU-T X.509], is utilized). This results in the production of the **ds:KeyInfo** and **wsse:BinarySecurityToken element** information items.

A.2.2.14 The signing application canonicalizes the **ds:SignedInfo element** information item to produce a sequence of octets (to be input to the signature method) as follows:

a) the input to the "exclusive canonical Fast Infoset algorithm without comments" is the XPath node set of the **ds:SignedInfo element** information item and its child information items (see 6.1.5 a);

b) a canonical XML document is produced from the XPath node set (see 6.1.5 a) using the canonical XML algorithm specified in W3C Exclusive Canonical XML (see 6.4.4);

c) the canonical XML document is parsed to produce an XML infoset (see 6.1.5 b);

d) the canonical XML infoset is serialized as a canonical fast infoset document (see 6.1.5 c), with the restrictions on serialization specified in 6.3; and

e) the sequence of octets that is input to the RSA-SHA-1 signature algorithm is the canonical fast infoset document.

A.2.2.15 The signing application generates the signature value by applying the signature method to the resulting octets produced from the canonicalization of the signed information. This results in the production of the **ds:SignatureValue element** information item. The element content of the **ds:SignatureValue element** information item will contain the characters that are the base64 (see [IETF RFC 2045]) encoding of the octets of the signature value generated by the RSA-SHA1 signature algorithm.

A.2.2.16 The signing application generates the signature. This results in the production of the **ds:Signature element** information item that contains, as child information items, the previously produced **ds:SignedInfo**, **ds:SignatureValue** and **ds:KeyInfo element** information items.

A.2.2.17 Finally, the signing application generates the Web services security SOAP header block. This results in the production of the **wsse:Security element** information item that contains, as child information items, the previously produced **wsse:BinarySecurityToken** and the **ds:Signature element** information items.

A.2.3 Validating the signed SOAP message infoset

A.2.3.1 The process of signature validation is specified in W3C XML Signature, 3.2. A description of this process as applied to the example signed SOAP message infoset follows.

A.2.3.2 The core validation process requires validation of references and validation of the signatures.

A.2.3.3 Before reference validation is performed, the validating application is required to canonicalize the signed information and operate on the canonicalized signed information. All references in the following subclauses to information items within the signed information, represented by the **ds:SignedInfo element** information item and contents, refer to the information items represented in canonical signed information.

NOTE – It is important that persons and automated mechanisms operate on the data that was transformed to be signed and not operate on the original pre-transformed data, see W3C XML Signature, 8.1.3.

A.2.3.4 The validating application obtains the algorithm, the "exclusive canonical Fast Infoset algorithm without comments", for canonicalizing the signed information, represented by **[normalized value]** property of the **Algorithm attribute** information item on the **ds:CanonicalizationMethod element** information item.

A.2.3.5 The validating application canonicalizes the signed information as follows:

a) the input to the "exclusive canonical Fast Infoset algorithm without comments" is the XPath node set of the **ds:SignedInfo element** information item and its child information items (see 6.1.5 a);

b) a canonical XML document is produced from the XPath node set (see 6.1.5 a) using the canonical XML algorithm specified in W3C Exclusive Canonical XML (see 6.4.4);

c) the canonical XML document is parsed to produce an XML infoset (see 6.1.5 b);

d) the canonical XML infoset is serialized as a canonical fast infoset document (see 6.1.5 c), with the restrictions on serialization specified in 6.3; and

e) the canonical fast infoset document is parsed to produce an XML infoset from which the signed information, represented by **ds:SignedInfo element** information, is obtained.

A.2.3.6 Next, the validating application determines that for the validation of references there is one reference to validate, represented by **ds:Reference element** information item.

A.2.3.7 The validating application obtains the data object to be digested by dereferencing the URI "**#TheBody**", which is represented by the **[normalized value]** property of the **URI attribute** information on the **ds:Reference element** information item, and executing the transforms, represented by the **ds:Transforms element** information item, on the data object.

A.2.3.8 Dereferencing is achieved by searching the SOAP message infoset to identify an **element** information item that has a **wsu:Id attribute** information item with a **[normalized value]** property equal to the identifier "**TheBody**". In this case, the XPath node set of the **soap:Body element** information item and its child information items is identified as is the data object to be digested.

A.2.3.9 The transforms consist of a single transform, represented by the **ds:Transform element** information item, that specifies the "exclusive canonical Fast Infoset algorithm without comments" (see 6.4.4 and clause 7), represented by the **Algorithm attribute** information item (on the **ds:Transform element** information item) whose **[normalized value]** property is "**urn:fastinfoset:c14n:exclusive**".

A.2.3.10 The validating application executes the transforms, resulting in the transformation from the data object to a sequence of octets (that are input to the digest algorithm), as follows:

a) the input to the "exclusive canonical Fast Infoset algorithm without comments" is the XPath node set of the **soap:Body element** information item and its child information items of the SOAP message infoset in A.1.4 (see 6.1.5 a);

b) a canonical XML document is produced from the XPath node set (see 6.1.5 a) using the canonical XML algorithm specified in W3C Exclusive Canonical XML (see 6.4.4);

c) the canonical XML document is parsed to produce an XML infoset (see 6.1.5 b);

d) the canonical XML infoset is serialized as a canonical fast infoset document (see 6.1.5 c), with the restrictions on serialization specified in 6.3; and

e) the sequence of octets that is input to the digest algorithm is the canonical fast infoset document.

A.2.3.11 The validating application digests the sequence of octets using the SHA-1 digest algorithm, represented by **ds:DigestMethod element** information item, to produce the digest value.

A.2.3.12 The validating application compares the produced digest value against the digest value in the reference, represented by the contents of the **ds:DigestValue element** information item. If there is any mismatch, then verification has failed.

A.2.3.13 Next, the validating application validates the signature.

A.2.3.14 The validating application obtains the keying information, represented by the **ds:KeyInfo** and **wsse:BinarySecurityToken element** information items (in this example an X.509 security token, see [ITU-T X.509], is utilized).

A.2.3.15 The validating application signs the sequence of octets that is the canonical fast infoset document obtained in A.2.3.5 using the RSA-SHA-1 signature algorithm, represented by **[normalized value]** property of the **Algorithm attribute** information item on the **ds:SignatureMethod element** information item, to produce a signature value.

A.2.3.16 Finally, the validating application confirms that the produced signature value is the same as the signature value in the signed information, represented by the contents of the **ds:SignedValue element** information item.

A.3 Encrypting and decrypting the SOAP message infoset

A.3.1 The signed and encrypted SOAP message infoset

A.3.1.1 The signed and encrypted SOAP message infoset of the SOAP message infoset in A.1.4 where the SOAP body has been signed and then the SOAP body contents have been encrypted is presented in Annex C.

A.3.2 Generating the signed and encrypted SOAP message infoset

A.3.2.1 First, the SOAP message infoset is signed, as described in A.2.2, and the encrypted SOAP message infoset (see Annex C) will be produced by adding information items to and replacing information items of the signed SOAP message infoset (see Annex B).

A.3.2.2 The process of encryption is specified in W3C XML Encryption, 4.1. A description of this process as applied to the signed SOAP message infoset follows.

A.3.2.3 The encrypting application selects one or more data items to be encrypted and for each data item: selects an algorithm to encrypt the data item; encrypts the data item; produces encrypted type information encapsulating the result of encrypting the data item (the cipher data); and replaces the data item with the encrypted type information.

A.3.2.4 The encrypting application selects that the single data item to be encrypted is the **n:payment element** information item (and content) of the SOAP message infoset in A.1.4.

A.3.2.5 The encrypting application selects that the RSA v1.5 Key Transport algorithm (see [IETF RFC 3447]) will be used for encryption key transportation and selects the same key that was used to sign the signature information (see A.2.2.13) as the key to use for encrypting the data item. This results in the production of the **xenc:EncryptionMethod**, **ds:KeyInfo** and **xenc:CipherData element** information items associated with the encrypted key information (see the **xenc:EncryptedKey element** information item).

A.3.2.6 The encrypting application selects that the triple DES algorithm (see [ANSI X9.52]) will be used for encrypting the **n:payment element** information item (and content). This results in the production of the **xenc:EncryptionMethod element** information item associated with the encrypted type information (see the **xenc:EncryptedData element** information item).

A.3.2.7 The encrypting application encrypts the **n:payment element** information item (and content) as follows:

a) the **n:payment element** information item shall be converted to a complete XML infoset (see 8.1.3 a);

b) that XML infoset shall be serialized to a fast infoset document (see 8.1.3 b); and

c) the octets of that fast infoset document are encrypted to produce an encrypted octet sequence (see 8.1.3 c).

A.3.2.8 The encrypting application generates encrypted type information. This results in the production of **xenc:EncryptedData element** information item that has the following:

a) a **wsu:Id attribute** information item with a **[normalized value]** property of "**EncryptedBodyContents**" that is the identifier of the encrypted data information;

b) a **Type attribute** information item with a **[normalized value]** property of "**urn:fastinfoset:element**" that identifies the data item that was encrypted is a fast infoset document generated from an **element** information item (see 8.1.4);

c) the child **xenc:EncryptedKey element** information item produced in A.3.2.6;

d) a child **xenc:CipherData element** information item that has a child **xenc:CipherValue element** information item whose contents are characters that are the base64 (see [IETF RFC 2045]) encoding of the encrypted sequence of octets produced in A.3.2.7 c.

A.3.2.9 The encrypting application generates the encrypted key information. This results in the production of **xenc:EncryptedData element** information item that has the following:

a) the child **xenc:EncryptionMethod**, **ds:KeyInfo** and **xenc:CipherData element** information items produced in A.3.2.5; and

b) a child **xenc:ReferenceList element** information item that has a child **wsse:Reference element** information item with a **URI attribute** information item whose **[normalized value]** property is the URI "**# EncryptedBodyContents**" that references the **xenc:EncryptedKey element** information item.

A.3.2.10 Finally, the encrypting application replaces the **n:payment element** information item (and content) with the produced **xenc:EncryptedData element** information item, as specified in 8.1.5, and the **xenc:EncryptedKey element** information item is added as a child of the **wsse:Security element** information item, and that child occurs before the **ds:Signature element** information item.

NOTE – The order of the **xenc:EncryptedData** and **ds:Signature** corresponds to the order in which encryption and signing are performed.

A.3.3 Verifying and decrypting the signed and encrypted SOAP message infoset

A.3.3.1 The process of decryption is specified in W3C XML Encryption, 4.2. A description of this process as applied to the encrypted and signed SOAP message infoset follows.

A.3.3.2 First, the decrypting application processes the encrypted key information, represented by the **xenc:EncryptedKey element** information item, to obtain the key to use for decrypting and the data that was encrypted using that key.

A.3.3.3 The decrypting application utilizes the RSA v1.5 Key Transport algorithm (see [IETF RFC 3447]) for encrypted key transportation and utilizes the same key that was used to verify the signature information (see A.2.3.14) as the key for decrypting the data.

A.3.3.4 The decrypting application identifies that the data that is to be decrypted is the **xenc:EncryptedData element** information item (that is referenced by the encrypted key information).

A.3.3.5 The decrypting application decrypts the octet sequence, which is represented by the base64 (see [IETF RFC 2045]) decoding of the characters of the **xenc:CipherValue element** information item, to obtain a cleartext octet sequence.

A.3.3.6 The type of the cleartext octet sequence is a fast infoset document generated from an **element** information item, represented by the **Type attribute** information item with a **[normalized value]** property of "**urn:fastinfoset:element**" (see 8.3 a).

A.3.3.7 The decrypting application obtains the **element** information item from the fast infoset document and replaces the **xenc:EncryptedData element** information item with that **element** information item (see 8.3 b and 8.3 c).

A.3.3.8 Finally, the signed and decrypted SOAP message infoset is validated as described in A.2.3.